

RM

(Relative Mark)

Name

Relative Line and Mark

Format

"RM n Δx1, Δy1, Δx2, Δy2, ..., Δxn, Δyn" ▽

Function

To draw a series of lines between a number of points plotting a specified mark at each point, where the points are specified as coordinates relative to one another.

Parameter type and range

The variables "n" specifies the mark according to the table shown in the "AM" command in the same range 0 to 15.

The size and rotation of the mark can be specified using the "SI" and "DI" commands respectively.

The marks are placed at the coordinates specified by the relative coordinates. The first mark is made at the point whose coordinates are (Δx1, Δy1) to the current pen position. The next point is marked at a relative coordinate to the first marked point, and so on. A line is drawn between each point, with a line type corresponding to the setting of the "LT" command.

As with the "AM" command, if no coordinates are given, the mark is placed at the current pen location.

ASCII Code Sequence

Decimal

(65)_D (82)_D (N1) (N2) (44)_D (X1) (X2)...
(Xn) (44)_D (Y1) (Y2)... (Yn) (44)_D (x1) (x2)...
(x3) (44)_D (y1) (y2)... (yn) (13)_D

or (65)_D (82)_D (N1) (N2) (44)_D (X1) (X2)...
(Xn) (44)_D (Y1) (Y2)... (Yn) (44)_D (x1) (x2)...
(x3) (44)_D (y1) (y2)... (yn) (00)_D

Hexadecimal

<41>_H <52>_H <N1> <N2> <2C>_H
<X1> <X2>... <Xn> <2C>_H <Y1>
<Y2>... <Yn> <2C>_H <x1> <x2>...
<x3> <2C>_H <y1> <y2>...
<yn> <0D>_H

or <41>_H <52>_H <N1> <N2> <2C>_H
<X1> <X2>...
<Xn> <2C>_H <Y1> <Y2>...
<Yn> <2C>_H <x1> <x2>... <x3>
<2C>_H <y1> <y2>... <yn> <00>_H

Where (N1) and (N2) are the ASCII codes for the characters of "n" and (X1)... (Xn) and (Y1) ... (Yn) are the codes for the characters of the coordinates of the first point specified as a decimal number, and (xn), (yn) of the next point and so on.

4.7 Error Handling Command

IM

(Input Mask)

Name Input mask

Format "IM e,s" ▽

Function To set up an error handling procedure.

Types of errors handled by the "IM" procedure.

- Command error

This error occurs when a command does not occur in the HI-80 command list for the particular plotting mode being used.

For example giving the command "Y 100,100" in either plotter mode 0 or plotter mode 1 will generate this error. Also giving a command from plotter mode 0 when plotter mode 1 is set will generate such an error and vice versa.

- Parameter range error.

This error will occur if the parameter given is outside the range allowed. For example in plotter mode 0 the "MA" command can have values for either parameter in the range -16383 to +16383. If the command "MA 155, 19000" is given, the parameter range error will occur.

- Missing parameter error.

If a parameter is missed off the end of a command, because such a parameter can be interpreted as zero, no error will be generated. However, if more than one is left out, the missing parameter error will occur. For example the "CA" command requires the position of the centre, the radius and the start and ending angles. Giving the command "CA 100,100" would generate this error because the radius and angles are missing.

- Delimiter error.

If a character other than a space or a comma occurs in a command and that character is not part of the syntax, then the incorrect delimiter error will be generated. This is particularly likely to occur if the character "E" is generated as part of a numerical variable and printed as such. Thus the BASIC command LPRINT "DA "; X; ",2002" will generate the error if the value of X given by the BASIC is a number expressed in scientific notation e.g. 3.45678E04.

- Overflow error.

This error will be generated if the result of a calculation exceeds the allowed range (-32768 to 32767). This occurs when trying to carry out a relative move and the pen position goes outside the allowable range. It can also occur if an attempt is made to divide by zero in a command such as "IF 1,0". Even if this error is masked as shown below, it may not be possible to continue with subsequent commands.

Parameter range and type

The parameter "e" allows the type of error to be selected.

The parameter "s" specifies how the error is handled when it is encountered.

The error handling applies to both plotter mode 0 and plotter mode 1, but can only be set in plotter mode 0. The printer mode will ignore errors and continue with the next part of the text.

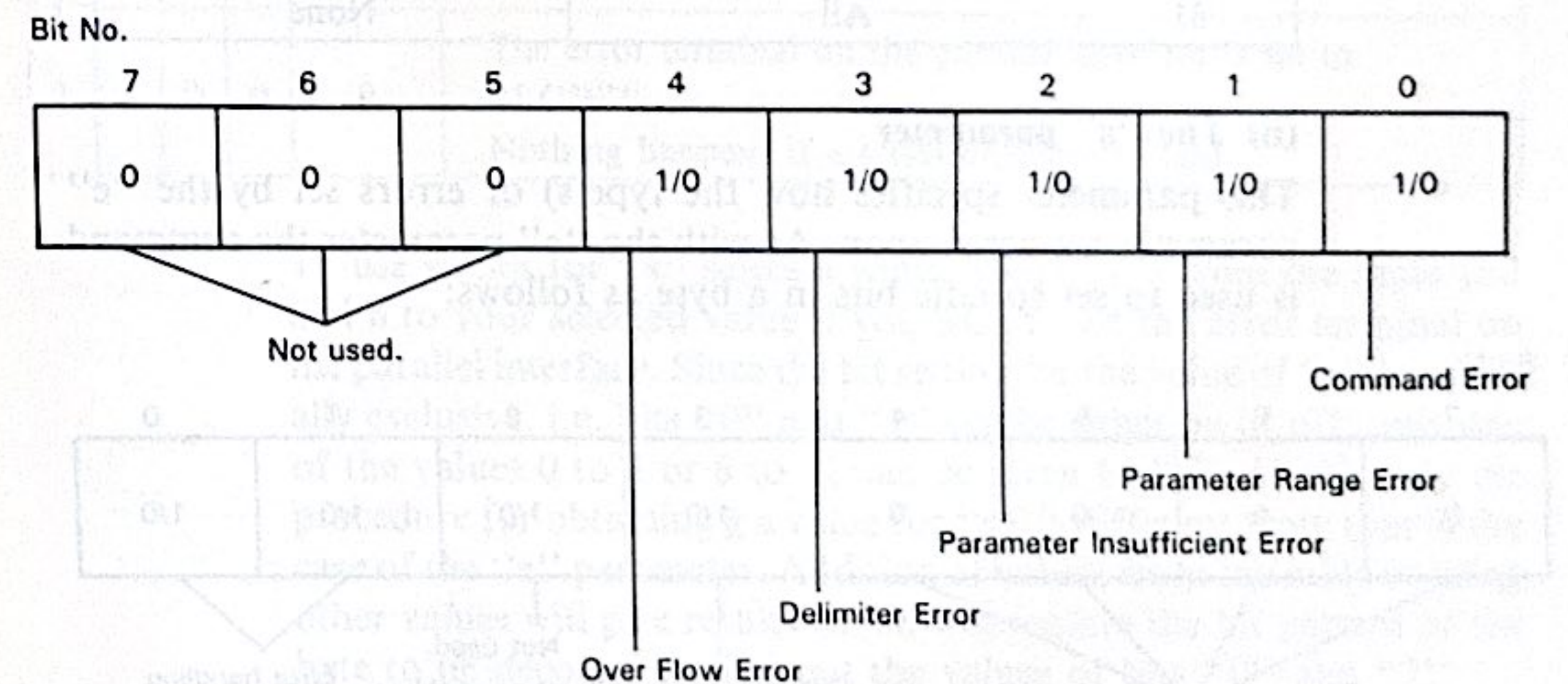
Other errors can arise which arise from other factors. These are discussed at the end of this section.

The characters of both parameters are converted into a single byte by the HI-80 software. This means they must be within the range 0 to 255.

(i) The "e" parameter

The value of this parameter decides which error condition will be acted upon. What type of action is taken depends on the value of the parameter "s".

The error handling is controlled by setting a particular bit of the "e" byte as follows:



When the bit is set to "0", the error handling procedure specified by the value of "s" is not performed. This means that if an error is encountered, the command containing it is ignored, the next command is carried out if it does not contain an error, and the error LED does not light.

When the bit is set to "1" the error handling specified by "s" is carried out. The command containing the error cannot be carried out.

To obtain a value for "e" use the following table, adding the values in the right hand column if you require the error in the left hand column to be taken into account.

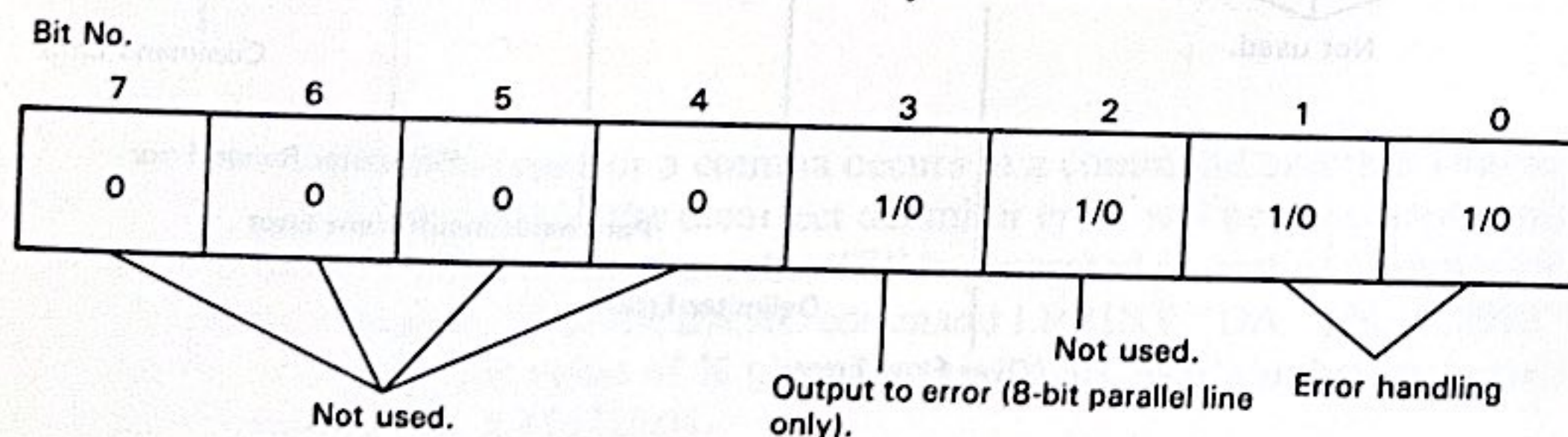
ERROR TYPE	VALUE
Command Error	1
Parameter Range Error	2
Parameter Insufficient error	4
Delimiter error	8
Overflow error	16

The following examples of values of "e" show how this table is used:

Value of "e"	error set	parameters ignored
6	Parameter range Parameter insufficient	Command Delimiter Overflow
17	Overflow Command	Parameter range Parameter insufficient Delimiter
31	All	None

(ii) The "s" parameter

This parameter specifies how the type(s) of errors set by the "e" parameter are acted upon. As with the "e" parameter the command is used to set specific bits in a byte as follows:



Two bits are used to give four possible means of controlling the error handling. A further bit is used to determine whether the HI-80 sets the error terminal on a parallel interface to stop further commands being sent.

The appropriate bit settings are described in the following table. To obtain a value for "s" use the number in the "VALUE" column.

BIT NUMBER				VALUE	ERROR HANDLING PROCEDURE
3	2	1	0		
0	0	0	0	0	The error LED is lit. The next command is executed, skipping the command containing the error. The error light remains lit to show an error has occurred and must be cleared with an error releasing command.
0	0	0	1	1	The error LED is lit and further input commands are not executed until an error releasing command is input.
0	0	1	0	2	The error LED is lit. The error number together with the command containing the error are printed in the lower left of the maximum effective drawing area. The next command is then executed. The error LED remains lit to show an error has occurred.
0	0	1	1	3	The error LED is lit. The error number together with the command containing the error are printed in the lower left of the maximum effective drawing area. The HI-80 waits for the input of an error releasing command.
1	0	0	0	8	The error terminal on the parallel interface is set to "LOW". Nothing happens if a serial interface is used.

To use values for "s" select a value from 0 to 3 from the table and add 8 to your selected value if you wish to set the error terminal on the parallel interface. Since the bit setting for the value of "s" is mutually exclusive, i.e. bits "0" and "1" can be either on or off, only one of the values 0 to 3 or 8 to 11 can be given to "s". In this way the procedure for obtaining a value for "s" is different from that in the case of the "e" parameter. Addition of values from the table or using other values will give results which will require the bit pattern of the byte to be decoded to find out the values of bits "0" and "1".

Example

The following sequence of commands demonstrate an error and show the resulting printout.

```
LPRINT "IM 31,3"
LPRINT "MA 1200,900"
LPRINT "DR"
```

The first command sets up the error handling with the "IM" command to detect all errors, print them out and wait for an error releasing command. The error occurs with the "DR" command which does not have any parameters. The error LED comes on and the message is printed as follows:

```
PARAMETER INSUFFICIENT ERROR!! CMD:dr
```

Up to 10 characters of the command are printed. The message is printed in the normal characters.

If the message has been printed and parameter "s" is set to allow the next command to continue (i.e. "s" is set to the value "2"), the pen will return to its position before the erroneous command was received.

Commands to release the error state

The following commands will release the error state if the HI-80 is operating in plotter mode 0.

COMMAND	NAME	INPUT BUFFER	PARAMETERS
HO	Home	remains	remain
CI	Clear Interface	cleared	remain
IN	Initialize	cleared	cleared
DF	Default	remains	cleared

Note:

Remember that the "DF" and "IN" commands will reset the parameters to the power up state, so that if the DIP switch is set to give plotter mode 1, these commands will set that mode. Also they will set the error handling procedure to the default, which is "IM 31,1".

The following commands will release the error state if the HI-80 is operating in mode 1.

COMMAND	NAME	INPUT BUFFER	PARAMETERS
H	Home	remains	remain
:	Clear Interface	cleared	remain
:	Clear	remains	cleared

NOTE:

Remember that the ":" command will reset the parameters to the power up state, so that if the DIP switch is set to give plotter mode 0, this commands will set that mode and a further error will be generated if further mode 1 commands are sent. Also the error handling will be set back to "IM 31,1" as default.

Types of errors not handled by the "IM" procedures

Errors can arise for a number of reasons other than the ones which can be set by the "e" parameter. They are mostly problems of BASIC or another language or the computer's operating system. Some of them are summarised as follows.

- A general method of finding the source of the problem.

Since the main function of the HI-80 is as a plotter, the problems usually arise because the plotter modes commands are incorrect. The error can usually be detected by the "IM" routine, but even then the reason is not obvious. If the plotter is switched to printer mode (most conveniently by switching it on with the CAP/OFF switch held down) the characters can be seen as they arrive at the printer. The most common problems are incorrect format of the command and text, and carriage returns which you did not expect.

- Problems arising from BASIC and particular computers.

(i) MICROSOFT BASIC

MICROSOFT BASIC has the feature that it normally sends a carriage return and line feed after 80 characters. If you wish to use commands which involve long strings of characters (e.g. "VC", "VR", "DA" and "DR") some characters will appear to be lost. This is because when the 80 characters are sent by BASIC, a carriage return and line feed are sent which is interpreted as the terminator character. The next characters are the rest of the numerical data. Depending on how many characters have been sent, errors could arise because there are insufficient parameters, or because the original command was correct and the number and or delimiter sent next is an invalid command. This problem can be overcome in two ways. Firstly by using the

WIDTH LPRINT 255

command which allows an infinite number of characters to be sent. Alternatively the command parameters can be split into shorter batches. Where commands are absolute no problem should arise in doing this. The "VA" command will require that the last parameter of each batch be repeated as the first character of the next batch, otherwise there will be a gap in the curve.

MICROSOFT BASIC also has another feature which can cause problems in plotter mode. If a string is more than 80 characters long it will be printed on a new line. Thus if for example a string is built up of a set of statements for moving the pen or drawing a curve, and is preceded by a variable, the long string is printed on a new line generating an error. This is best illustrated by the following example.

LPRINT "DA 1200, "; P ; ", 200, 300,....., 900, 300"

the result would be that a carriage return is printed after the variable "P" and because the string is long, the string is then printed on a separate line if printed in text mode. If "P" had the value "900", the result in text mode would be:

**DA 1200, 900
,200, 300,.....etc.**

The plotter mode error would occur in this case when the new line is printed, since it does not have any command to prefix it.

Using the text mode in this case to see what has actually been sent to the HI-80 is obviously very useful.

(ii) APPLESOFT BASIC

There are a number of problems in using the HI-80 with APPLESOFT.

In using the APPLESOFT "TAB" command there are two problems. The first is that "TAB" is set for the normal 40 column APPLE screen. Instead of "TAB" use

POKE 36,n

where n is the column you wish to TAB to. First however set the column width to 80 using

CTRL - I 80N

or however else your interface manual tells you to set the column width.

In using the ASCII code (9)_D to TAB when the printer TAB settings have been set up by the <ESC> "D" command, the interface will normally ignore this character. This is because it is a CTRL-I which is normally used by the card as a means of initiating a procedure such as setting the column width or dumping the graphics screen in a particular way. Such interface cards usually have a means of changing this control character. Please consult your printer card interface manual to see how overcome this problem.

In text mode the ability to send ASCII codes over the full set will cause problems in trying to change character heights or other commands where ASCII codes have to be sent using the CHR\$() command. The APPLE interface will probably filter out characters such as CHR\$(9), which is CTRL-I and is used to control interface settings. Since there are as many problems as there are interface cards, please consult your dealer or APPLE USER GROUP to help you with your own particular problem.

(iii) TRS-80 BASIC

The main problems with TRS-80 BASICS are not being able to send a minority of the characters to the printer. This is best overcome by bypassing BASIC. By poking the character into location 14312 it can be sent to the printer directly. The status of the printer can be seen by testing this location to see if it contains the value "63". This will solve the problem on MODEL I and MODEL III.

(iv) BBC BASIC

The command to send the next character to the printer port is VDU 1. However in plotter mode, since everything goes to the printer as a text string, simply execute the command to switch the output to the printer port (VDU 2) and print the strings and variables. All examples in plotter mode can then be used with PRINT instead of LPRINT. When using the HI-80 in printer mode use the VDU 1 command for control sequences. The ASCII code sequences have been placed at the bottom of each page to help you in this respect. For example to set double width characters until switched off, the <ESC> "W" (1) sequence, use

VDU 1, 27, 1, 87, 1, 1

Remember a "1" has to be used before EACH character you wish to send. The VDU 3 command is used to turn off output to the printer.

(v) COMMODORE COMPUTERS

The Commodore series of computers uses a character set which is not the standard ASCII set. You will have to buy an interface which converts the characters to true ASCII, or use a special software printer driver to carry out the conversion.

(vi) ATARI COMPUTERS

There are various problems with the ATARI LPRINT statement. Use the text mode as described above to determine the cause of the problem. The best solution is to use PRINT #1 instead of LPRINT.

(vii) MF BASIC on the EPSON QX-10

If you cannot print control characters, which in practice means the <ESC> codes in text mode as there is only one control character in plotter mode 0, it is because you have not executed the BIT ON command first.

• Problem with a serial interface.

When communicating with a serial interface various problems can arise both because of the nature of the method of communication and the need to set up the correct protocol.

Some commands require a large amount of drawing time. The circle and the hatching a bar and a pie in particular require a large number of points and lines to be plotted. The serial interfaces inserted in the HI-80 and the serial interface of the computer may not allow communication to stop data being sent by the computer when the HI-80 is busy. In this case commands will continue being sent and the HI-80 will not be able to cope because it is busy calculating and plotting. Whereas there may be a buffer on the interface board in the HI-80, at some stage this buffer will overflow. An error will be generated because the characters in the buffer become corrupted in their order. The simplest way to overcome this is to put a delay loop in the program so that the buffer can keep up with the supply of data.

The other type of problems arise from the need to set the protocol correctly. If the rate of transmission, the parity and stop bits or the number of data bits being sent are not matched, data will be received in a corrupt state. Consult your dealer if you feel you have checked the computer and interface settings and the problem has not been resolved.

If the data bits are set to 7 it will only be possible to send ASCII codes 0 to 127. If some commands do not appear to work, check that you are sending 8 bit codes correctly. You can do this by printing characters in the text mode which have ASCII codes in the range 160 to 255. They should print as italic characters.

Chapter 5 PLOTTER MODE 1 COMMANDS

When the plotter is in Mode 1, it will accept commands from the host computer. The plotter will be in Mode 1 when the host computer sends the command `MODE 1` to the plotter. The plotter will be in Mode 1 when the host computer sends the command `MODE 1` to the plotter.

Parameter range

no parameters required

Example

`!PRINT`

ASCII Code Sequence

Decimal

ASCII Code

Hexadecimal

Hexadecimal

Hexadecimal

5.1 Configuration Commands

:

(Clear)

Name

Clear

Format

":":

Equivalent Mode 0 command

"IN"

Function

Printing this string causes the plotter to be returned to the default state for this mode. The buffer will be cleared. Even if the DIP switch has been set to place the HI-80 in plotter mode 0 on power up, it will stay in mode 1.

Parameter range

no parameters required

Example

LPRINT ":"

ASCII Code Sequence

Decimal

(58)_D (13)_D

or (58)_D (0)_D

Hexadecimal

<3A>_H <0D>_H

or <3A>_H <00>_H

(Interface Clear)

Name Interface Clear

Format “;” ▾

Equivalent Mode 0 command
“CI”

Function Printing this string causes the input buffer to be cleared and any error state to be released.

Parameters such as magnification, origin etc are not reset.

Parameter range
no parameters required

Example LPRINT “;”

ASCII Code Sequence

Decimal (59)_D (13)_D
or (59)_D (0)_D
Hexadecimal <3B>_H <0D>_H
or <3B>_H <00>_H

J

(Select Pen)

Name New pen

Format “Jn” ▾

Equivalent Mode 0 command
“SPn”

Function Selects a pen, or caps the four pens.

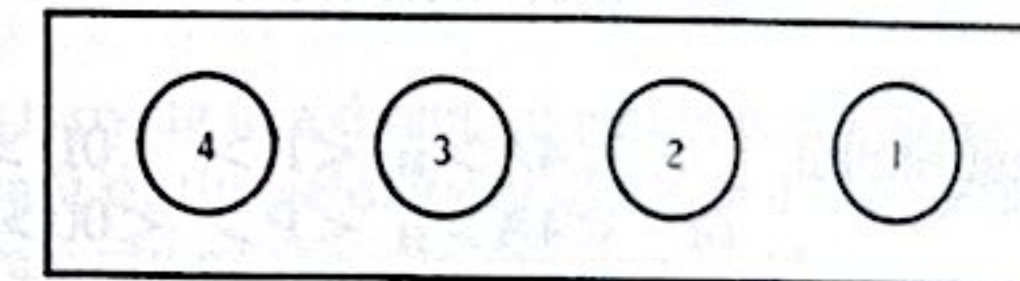
Parameter type and range

“n” is a character on the range -1 to 8.

Example LPRINT “J2”

This selects pen number 2.

The pen positions are marked on the pen holder and are as follows:



Pen number 1 is selected when the power supply is turned on. When a “Jn” command is executed, the pen is lifted and the pen carriage moved to the left or right when the pen is changed. If the pen number chosen is greater than the current one, the carriage moves to the right, if it is smaller it moves to the left. If the number is the same as the current pen number, no action is taken.

The pen is then returned to the coordinate position it was at when the “Jn” command was given.

When $n = 0$ or -1 the carriage moves to the right and the pens are stored and capped.

$n = 1$ or $n = 5$	pen 1 is selected
$n = 2$ or $n = 6$	pen 2 is selected
$n = 3$ or $n = 7$	pen 3 is selected
$n = 4$ or $n = 8$	pen 4 is selected

If $n < -1$ or $n > 8$ then an error is generated.

The "J0" command is equivalent to pressing the CAP/OFF key. The CAP/OFF key cannot be pressed if the ONLINE LED is lit. Operating the CAP/OFF key can only be done manually by pressing the ONLINE key first to switch the HI-80 offline. However, the "J0" command will cap the pens leaving the ONLINE LED switched on. The ONLINE switch then becomes inactive. Before any other software command can be executed, or any other keys can be used, the CAP/OFF key must be pressed.

On the other hand the "J-1" command will allow the pens to be capped without the necessity for the CAP/OFF switch to be pressed. If another command is given the pen will move to the position it was at before the "J-1" command was given.

ASCII Code Sequence

Decimal (74)_D (P) (13)_D
or (74)_D (P) (00)_D

Hexadecimal <4A>_H <P> <0D>_H
or <4A>_H <P> <00>_H

where (P) is the ASCII code for the numeric value of "n", i.e. (48)_D or <30>_H if "n" is "0", (49)_D or <31>_H if it is "1" and so on to (56)_D or <38>_H if it is "8".

T

(Pen speed)

Name Pen speed

Function To specify the pen drawing speed.

Format "Tn" ▽

Equivalent Mode 0 command

"VS n"

Parameter type and range

"n" is a character "0" or "1". If "n" is a character other than "0" or "1" then an error is generated.

Example

LPRINT "T1"

When $n = 0$ (default value) the pen velocity is 230 mm/sec

When $n = 1$ the pen velocity is 100 mm/sec

If the pen is moving in a direction which is not along the axes, the speed is the resultant of the velocity in the X and Y directions. For example, the linear velocity of the pen moving at 45 degrees to both axes is $\sqrt{2}$ times the specified velocity.

ASCII Code Sequence

• For the fast speed "T0"

Decimal (84)_D (48)_D (13)_D
or (84)_D (48)_D (00)_D

Hexadecimal <54>_H <30>_H <0D>_H
or <54>_H <30>_H <00>_H

• For the slow speed "T1"

(84)_D (49)_D (13)_D
or (84)_D (49)_D (00)_D

Hexadecimal <54>_H <31>_H <0D>_H
or <54>_H <31>_H <00>_H

AC

(Alternate Mode)

Name Alternate command mode

Format "AC n" ▽

Function Printing this string causes the plotter to change to an alternate command mode.

Parameter range

If the value of "n" is "0" the alternate plotting mode is selected. The same command is used in plotter mode 0 to switch back to plotter mode 1.

If a value of "n" less than "0" or greater than "1" is given, an error will be generated. Future options will include other plotting modes. In these cases, the error will only be generated if the option ROM is not sensed. Please consult the manual referring to that option for further details.

When changing between plotter modes all the parameters and buffer contents are retained. However, some parameters are not the same in the two modes. Appendix F describes changing between modes.

Note:

This is the only two letter command in mode 1.

Example LPRINT "AC 1"

ASCII Code Sequence

Decimal (65)_D (67)_D (13)_D
or (65)_D (67)_D (0)_D
Hexadecimal <41>_H <43>_H <0D>_H
or <41>_H <43>_H <0>_H

5.2 Pen Positioning Commands

M

(Absolute Move)

Name Absolute Move

Format "M x , y" ▽

Equivalent Mode 0 command

"MA x,y"

Function To lift the pen and move it to an absolute location. The pen is moved to the point with coordinate (x,y).

If a series of coordinates is given as part of the command, the pen will move from point to point without plotting.

Parameter type and range

X and Y are integer numerical strings in the range - 16383 to 16383 inclusive.

Examples

- (i) LPRINT "M 1200,900"
- (ii) LPRINT "M"; X% ; "," ; Y%
- (iii) LPRINT "M" ; X ; "," ; Y
- (iv) LPRINT "M" ; 1000 "," ; 200

The examples have the following results:

- (i) The coordinates are printed as part of the string.
- (ii) The coordinates are printed as integer variables. The individual characters of the numbers will be sent to the printer.
- (iii) The coordinates are variables. If the variables are not integers all characters after and including the decimal point will be ignored.
- (iv) The coordinates are numerical constants, and the individual characters will be sent to the printer.

ASCII Code Sequence

Decimal

(77)_D (X1) (X2)... (X_n) (44)_D (Y1) (Y2)...
(Y_m) (13)_D
or (77)_D (X1) (X2)... (X_n) (44)_D (Y1) (Y2)...
(Y_m) (0)_D

Hexadecimal

<4D>_H <X1> <X2>... <X_n> <2C>_H
<Y1> <Y2>... <Y_m> <0D>_H
or <4D>_H <X1> <X2>... <X_n> <2C>_H
<Y1> <Y2>... <Y_m> <00>_H

Where (X1)... (X_n) are the characters of the digits of the X coordinate in decimal, and (Y1)..... (Y_m) are the digits of the Y coordinate in decimal.

R**(Relative Move)****Name**

Relative Move

Format

"R Δx , Δy" ▽

Equivalent Mode 0 command

"MR Δx, Δy"

Function

To lift the pen and move it to a location relative to the current position of the pen. The pen is moved from the current point (coordinate x,y) to the point with coordinate (x + Δx , y + Δy).

If a series of coordinates is given as part of the command, the pen will move from point to point without plotting.

Parameter type and range

X and Y are integer numerical strings in the range - 16383 to 16383 inclusive. If the value of either coordinate of the destination point is outside the range - 32768 to 32767, an error will result.

Examples

- (i) LPRINT "R 200,300"
- (ii) LPRINT "R"; X% ; "," ; Y%
- (iii) LPRINT "R" ; X ; "," ; Y
- (iv) LPRINT "R" ; 100 "," ; 200

The examples show the different formats as follows:

- (i) The coordinates are printed as part of the string.
- (ii) The coordinates are printed as integer variables. The individual characters of the numbers will be sent to the printer.
- (iii) The coordinate are variables. If the values are not integers any characters after and including the decimal point will be ignored.
- (iv) The coordinates are numerical constants, and the individual characters will be sent to the printer.

ASCII Code Sequence

Decimal

(82)_D (X1) (X2)... (X_n) (44)_D (Y1) (Y2)...

(Y_m) (13)_D

or (82)_D (X1) (X2)... (X_n) (44)_D (Y1) (Y2)...

(Y_m) (00)_D

Hexadecimal

<52>_H <X1> <X2>... <X_n> <2C>_D

<Y1> <Y2>... <Y_m> <0D>_H

or <52>_H <X1> <X2>... <X_n> <2C>_D

<Y1> <Y2>... <Y_m> <00>_H

Where (X1)... (X_n) are the characters of the digits of the X coordinate in decimal, and (Y1)... (Y_m) are the digits of the Y coordinate in decimal.

H

(Home)

Name

Home position

Format

"H" ▾

Equivalent Mode 0 command

"HO"

Function

To lift the pen and position it at the coordinate origin.

The movement to the origin is equivalent to the command "M 0,0".

The position of the origin CANNOT be changed in mode 1.

If an error has occurred, execution of this command will cause the error to be reset and the pen will be moved to the origin.

Example

LPRINT "H"

ASCII Code Sequence

Decimal

(72)_D (13)_D

or (72)_D (00)_D

Hexadecimal

<48>_H <0D>_H

or <48>_H <00>_H

5.3 Line Drawing Commands

(Absolute Draw)

D

Name Absolute Draw

Format "D x , y" ▽ or
"D x1 , y1 , x2, y2 , xk , yk" ▽

Equivalent Mode 0 command

"DA x1,y1"

Function

To draw a line from the current pen position to an absolute location. The line is drawn to the location whose coordinate is (x,y).

If a number of points are specified with coordinates (x1,y1), (x2,y2).... (xk,yk), the pen will draw successive lines between the points denoted by the coordinates. There is no limit to the number of points the HI-80 can receive in this command.

Parameter type and range

The coordinates x, y, xk, yk are integer numerical strings in the range -16383 to 16383. Each coordinate represents a distance of 0.1 mm.

Examples

- (i) LPRINT "D 1200,900"
- (ii) LPRINT "D"; X%; "," ; Y%
- (iii) LPRINT "D"; X ; "," ; Y
- (iv) LPRINT "D"; 1000 "," ; 200
- (v) LPRINT "D 100, 100, 500, 300, 0, 0, 100, 100"
- (vi) LPRINT "D"; X1%; "," ; Y1%; "," ; X2%; "," ; Y2%

The examples show the different formats as follows:

- (i) The coordinates are shown as part of the command string. The pen would draw from the current pen location to the coordinate (1200,900).

- (ii) The coordinates are given as integer variables. The digits will be sent to the printer as individual characters, and the pen will draw to the point whose coordinates are (X%,Y%), from the current pen location.
- (iii) The coordinates are fixed point variables. If they are not integers any characters after and including the decimal point will be ignored.
- (iv) The values are numerical constants and the individual characters will be sent to the printer.
- (v) A number of points are given inside the string, and the pen will draw from one point to the next.
- (vi) This shows the structure of a command to draw from the current pen position to two points using integer variables.

ASCII code Sequence

Decimal

(68)_D (x1) (x2)... (xn) (44)_D (y1) (y2)... (yn)
(44)_D (X1) (X2)... (Xn) (44)_D (Y1) (Y2)....
(Yn) (13)_D

or (68)_D (x1) (x2)... (xn) (44)_D (y1) (y2)... (yn)
(44)_D (X1) (X2)... (Xn) (44)_D (Y1) (Y2)....
(Yn) (0)_D

Hexadecimal

<44>_H <x1> <x2>... <xn> <2C>_H
<y1> <y2>... <yn> <2C>_H <X1>
<X2>... <Xn> <2C>_H <Y1> <Y2>....
<Yn> <0D>_H

or <44>_H <x1> <x2>... <xn> <2C>_H
<y1> <y2>... <yn> <2C>_H <X1>
<X2>... <Xn> <2C>_H <Y1> <Y2>...
<Yn> <00>_H

Where (x1)...(xn) and (y1) ... (yn) are the ASCII codes for the characters of the first coordinate in decimal, (X1)... (Xn) and (Y1) ... (Yn) for a subsequent one and so on.

(Draw Relative)

Name

Draw Relative

Format

"I Δx , Δy " ▽ or
"I Δx_1 , Δy_1 , Δx_2 , Δy_2 , ..., Δx_k , Δy_k " ▽

Equivalent Mode 0 command

"DR Δx , Δy "

Function

To draw a line from the current pen position to a position whose coordinates differ by the values of ($\Delta x, \Delta y$).

If a number of points are specified with relative coordinates ($\Delta x_1, \Delta y_1$), ($\Delta x_2, \Delta y_2$)... ($\Delta x_k, \Delta y_k$), the pen will draw successive lines between the points denoted by the coordinates beginning with the current pen location. Each movement will be relative to the pen position of the previous plotted point in the sequence. There is no limit to the number of points the HI-80 can receive in this command.

Parameter type and range

The coordinates Δx , Δy , Δx_k , Δy_k are integer numerical strings in the range -16383 to 16383.

If the absolute location of any plotted point is outside the range -32768 to 32767, an error will be generated.

Examples

- (i) LPRINT "I 300,200"
- (ii) LPRINT "I"; X%; ", " ; Y%
- (iii) LPRINT "I"; X ; ", " ; Y
- (iv) LPRINT "I"; 100 ", " ; 60
- (v) LPRINT "I 100, 100, 30, 150, 65, 100"
- (vi) LPRINT "I"; X1%; ", " ; Y1%; ", " ; X2%; ", " ; Y2%

The examples have the following results:

- (i) The coordinates are shown as part of the command string. The pen would draw from the current pen location to the coordinate (x_0, y_0) to the position ($x_0 + 300, y_0 + 200$).

- (ii) The coordinates are given as integer variables. The digits will be sent to the printer as individual characters, and the pen will draw to the point whose coordinates are ($x_0 + X\%$, $y_0 + Y\%$), from the current pen location with coordinates (x_0, y_0).
- (iii) The coordinates are fixed point variables. If they are not integers all characters after and including the decimal point are ignored.
- (iv) The values are numerical constants and the individual characters will be sent to the printer.
- (v) A number of points are given inside the string, and the pen will draw from one point to the next. If the pen position when the command is given is (x_0, y_0) the lines will be drawn from (x_0, y_0) to ($x_0 + 100, y_0 + 100$) to ($x_0 + 130, y_0 + 250$) to ($x_0 + 195, y_0 + 350$).
- (vi) This shows the structure of a command to draw from the current pen position to two points using integer variables.

ASCII Code Sequence

(73)_D (x1) (x2)... (xn) (44)_D (y1) (y2)... (yn)
(44)_D (X1) (X2)... (Xn) (44)_D (Y1) (Y2)...
(Yn) (13)_D
or (73)_D (x1) (x2)... (xn) (44)_D (y1) (y2)... (yn)
(44)_D (X1) (X2)... (Xn) (44)_D (Y1) (Y2)...
(Yn) (0)_D

Hexadecimal

<49>_H <x1> <x2>... <xn> <2C>_H
<y1> <y2>...
<yn> <2C>_H <X1> <X2>... <Xn>
<2C>_H <Y1> <Y2>... <Yn> <0D>_H
or <49>_H <x1> <x2>... <xn> <2C>_H
<y1> <y2>...
<yn> <2C>_H <X1> <X2>... <Xn>
<2C>_H <Y1> <Y2>... <Yn> <00>_H

Where (x1)...(xn) and (y1) ... (yn) are the ASCII codes for the characters of the first coordinate in decimal, (X1)... (Xn) and (Y1) ... (Yn) for a subsequent one and so on.

L

(Line Type)

Name

Line type.

Format

"Lp" ▾

Equivalent Mode 0 command

"LT p"

Function

To set the type of line drawn. The line may be solid, broken, dotted or a mixture.

The spacing of the various types is set with the "B" command.

The "L" command alters the line drawn by the following commands:

D (Absolute Draw)

I (Relative Draw)

Parameter type and range

"p" is a numerical character in the range 0 to 8 inclusive. If the character "p" is "9" or any other character, an error will be generated.

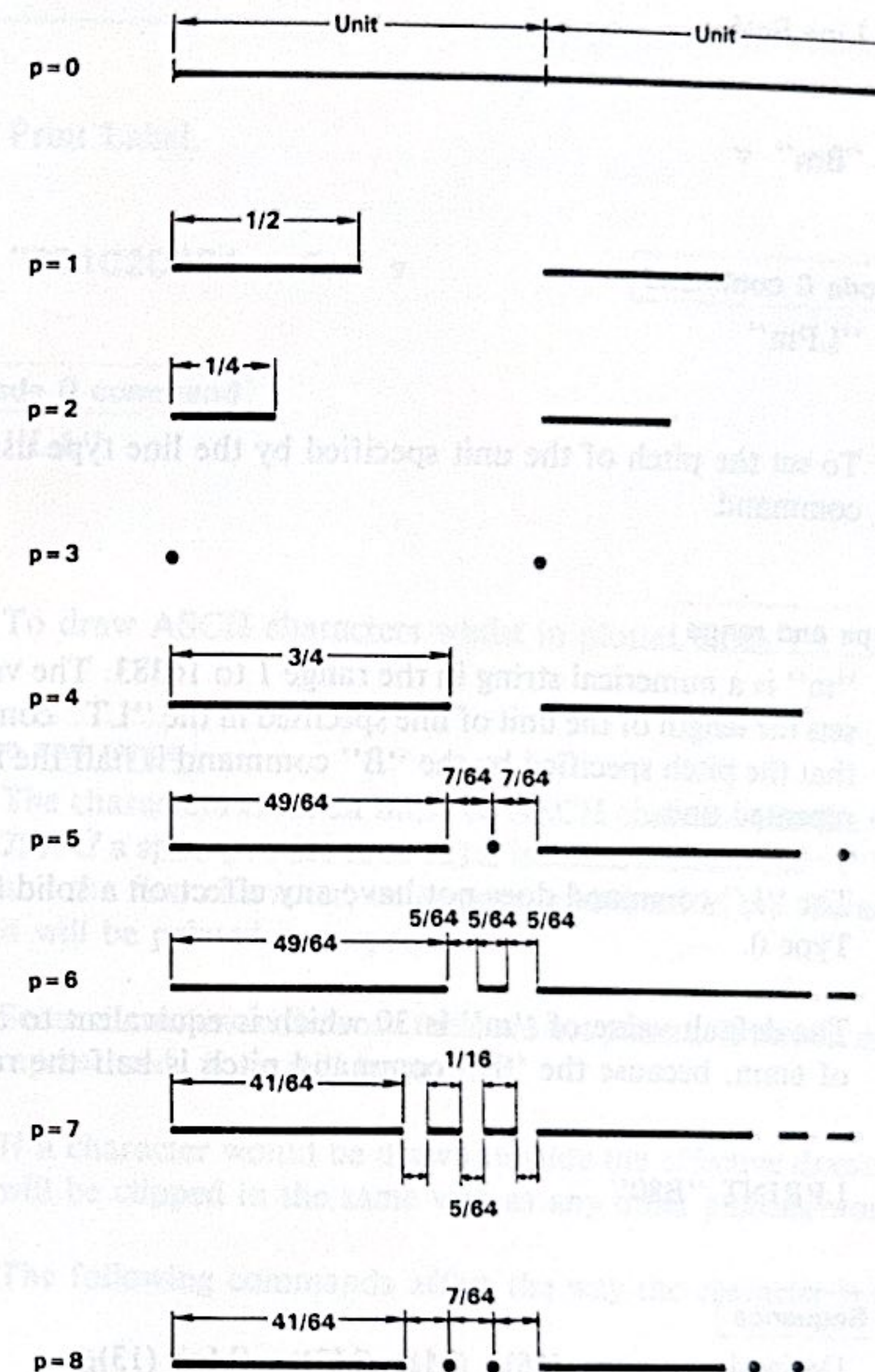
Examples

LPRINT "L3"

This draws a dotted line. The lines produced are as follows:

value of p	example
0	_____
1	-----
2	-----
3	
4	-----
5	
6	-----
7	
8	-----

Each line is drawn in repeated units. The length of a unit is set with the "B" command. The subdivision of each unit for the different styles is as follows:



ASCII Code Sequence

Decimal (76)_D (P) (13)_D
 or (76)_D (P) (0)_D
 Hexadecimal <4C>_H <P> <0D>_H
 <4C>_H <P> <00>_H

where (P) is the ASCII code for the numeric value of "p", i.e. (49)_D or <30>_H if "p" is "0", (49)_D or <31>_H if it is "1" and so on to (56)_D or <38>_H if it is "8".

B

(Line Scale)

Name

Line Scale.

Format

"Bm" ▾

Equivalent Mode 0 command

"LPm"

Function

To set the pitch of the unit specified by the line type using the "L" command.

Parameter type and range

"m" is a numerical string in the range 1 to 16383. The value of "m" sets the length of the unit of line specified in the "LT" command. Note that the pitch specified by the "B" command is half the length of the repeated unit.

The "L" command does not have any effect on a solid line, ie. Line Type 0.

The default value of "m" is 30 which is equivalent to a unit length of 6mm, because the "B" command pitch is half the repeat unit.

Example

LPRINT "B80"

ASCII Code Sequence

Decimal (65)_D (M1) (M2)... (Mn) (13)_D
or (65)_D (M1) (M2)... (Mn) (00)_D

Hexadecimal <42>_H <M1> <M2>... <Mn> <0D>_H
or <42>_H <M1> <M2>... <Mn> <00>_H

Where M1, M2,... Mn are the ASCII codes for the characters corresponding to the digits of the value of "m", when "m" is expressed as a decimal number.

5.4 Character Drawing Commands

P

(Label)

Name

Print Label.

Format

"PC1C2C3C4.....Cn" ▾

Equivalent Mode 0 command

"LA"

Function

To draw ASCII characters whilst in plotter mode 1.

Parameter type and range

The characters C1...Cn must be ASCII characters in the range 32 to 255. If a space(ASCII code 32) is inserted between the "P" command and the first character, or a space is inserted in the character string, it will be printed as a space.

Some characters in the code table are not present. If there is no character represented, none will be printed.

If a character would be drawn outside the effective drawing area, it will be clipped in the same way as any other plotting would be.

The following commands affect the way the character is drawn:

S specifies the size of the character
Q specifies the direction and size of rotation

These parameters can be reset using the "A" command. A table of default settings is given under the "A" command.

It is only possible to make special marks on a line as part of the "N" command.

When a character is printed the cursor is left at the last plotted position, so a carriage return does not move onto the 'next line'. There is no 'next line' as there would be when printing text. A carriage return is simply a termination command in plotter mode 0. The only way to move the cursor is with an "M" command.

Example

```
10 LPRINT "PAlways read the manual"
20 LPRINT "Pif you have a problem."
```

would print:

Always read the manual if you have a problem.

beginning at the current cursor position according to the current settings of the parameters affecting the size rotation etc.

ASCII Code Sequence

Decimal (80)_D (A1) (A2)... (An) (13)_D
 or (80)_D (A1) (A2)... (An) (00)_D
 Hexadecimal <50>_H <A1> <A2>... <An> <0D>_H
 or <50>_H <A1> <A2>... <An> <00>_H

Where A1 ... An are the ASCII codes of the characters to be printed.

S

(Scale)

Name Alpha scale

Format "Sn" ▽

Equivalent command in Mode 0

"SI"

Function To specify the height and width of the characters printed by the "P" command, and marks made by the "M" command.

Parameter type and range

The variable "n" must be in the range 0 to 254. As with other commands, the numerical digits of the value of "n" are sent as the ASCII codes of the digits. The value of "n" defines the height of the character as a multiple of a basic unit of 0.7 mm in the case of ASCII characters plotted by the "P" command and 0.4 mm in the case of marks made by the "N" command.

The parameters affect the character height and width as follows:

(i) ASCII characters produced by the "P" command:

Character height = $0.7 \times (n+1)$ mm
 Character width = $2/3$ of character height
 Distance between start of each character = $0.7 \times (n+1)$ mm

(ii) Marks produced by "N"

height = $0.4 \times (n+1)$ mm
 width = $0.4 \times (n+1)$ mm

Note:

For $n = 0$ or $n = 1$ the size of the mark is the same.

Example

10 LPRINT "S 8"
 20 LPRINT "PThese are the new size letters."

would print the sentence 5.6mm high and 4.2mm wide, thus:

These are the new size letters.

ASCII Code Sequence

Decimal (83)_D (N1) (N2)... (Nn) (13)_D
 or (83)_D (N1) (N2)... (Nn) (0)_D

Hexadecimal <53>_H <N1> <N2>... <Nn>
 <0D>_H
 or <53>_H <N1> <N2>... <Nn>
 <00>_H

Where (N1)... (Nn) are the characters of the digits of the parameter "n" expressed as decimal numbers.

Q**(Alpha Rotate)**

Name Alpha Rotate

Format "Qn" ▽

Nearest Equivalent command in Mode 0
 "DI"

Function

To set the direction of rotation of characters for printing by the "P" command and of marks made by the "N" command.

Parameter type and range

The value of "n" must be within the range 0 to 3 inclusive.

The value of "n" specifies a rotation of the character in units of 90 degrees. By specifying the value as positive the rotation takes place in an anticlockwise direction. If the value is negative the rotation is clockwise. The rotation is measured from the X axis.

value of n	angle of rotation
0	0°
1	90°
2	180°
3	270°

This command is limited when compared with the "DI" command of mode 0. The "DI" command can specify the angle in fractions of a degree, whereas the "Q" command only allows rotation in multiples of a right angle.

ASCII Code Sequence

Decimal (81)_D (N) (13)_D
 or (81)_D (N) (00)_D

Hexadecimal <51>_H <N> <0D>_H
 or <51>_H <N> <00>_H

Where (N) is the ASCII code for the characters "0" to "3" for whichever rotation is required.

A

(Alpha Reset)

Name Alpha Reset

Format "A" ▽

Function

To reset the functions relating to the way characters are printed in mode 1 to their default states without resetting the drawing functions.

The characters printed by the commands "P" and "N" are set to the following default values. These values are the default values on power up, or by giving the ":" command in plotter mode 1.

COMMAND	NAME	DEFAULT	EFFECT
S	Alpha Scale	n = 3	Sets characters to 2.4 by 2.4 mm
Q	Alpha Rotate	n = 0	Rotation angle is 0°

ASCII Code Sequence

Decimal (65)_D (13)_D
or (65)_D (0)_D

Hexadecimal <41>_H <0D>_H
or <41>_H <00>_H

5.5 Graph Drawing Commands

X

Name

Axis

Format

"X p, q, r" ▽

Function

To draw a line parallel to either the X or Y axis and graduate it.

Parameter type and range

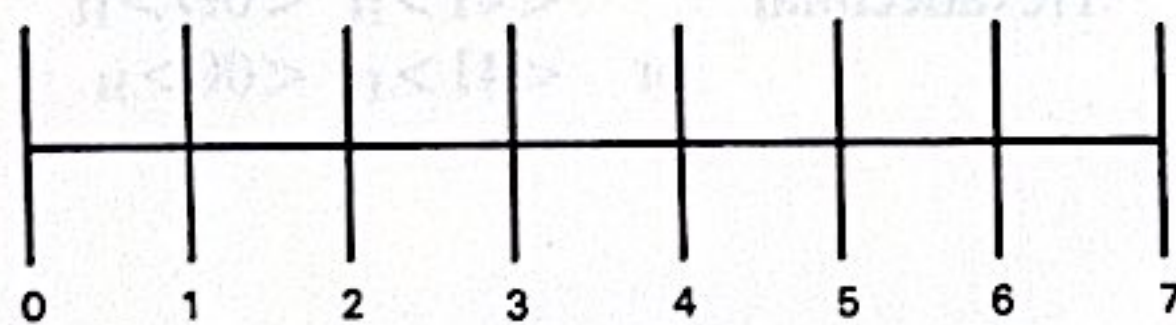
p specifies the direction of the axis and the type of parameter for q and as in table below.

p = 0 or 2 draws parallel to Y axis

p = 1 or 3 draws parallel to X axis

q specifies either the unit length of the divisions or the overall length of the line in the range -16383 to +16383, but MUST NOT be 0.

r specifies either the number of repetitions or the number of divisions, and is in the range 1 to 255.



The following table shows the parameter type of q and r when p takes different values.

p	AXIS	q	r
0	Y	Unit length	number of repetitions
1	X		
2	Y	Overall length	Number of Divisions
3	X		

When p = 0 or 1, the axis line is drawn and divided into an number of equal parts according to the value of "r". The ticks are then plotted.

When p = 2 or 3 the value of q/r (i.e. the overall length divided by the number of divisions) is used to find the unit length. If it is not an integer, the remainder is discarded. This results in a slightly shorter line. For example, if the command is given as follows:

LPRINT "X 2, 1000, 13, 0"

This should plot an axis which is 1000 mm long, with 13 tick marks. However since $1000/13 = 76.9231$, there is a considerable remainder which will be discarded. The unit length will therefore be 76 units of 0.1 mm which total $76 \times 13 \times 0.1 = 98.8$ mm. Thus the line is 1.2 mm shorter than the true figures.

To draw the axis in the negative direction, make "q" negative.

All characters are drawn according to the setting of the "A" (alpha scale).

The scale numbers are centred on the tick marks. However, if the character has an offset, this will move them from this position. Also if the character size is such that the height of the characters is greater than the unit length, the characters will overlap.

Both the ticks and the axis line will be drawn as a solid line regardless of the setting of the "L" (line type) command.

This command does not label the axes.

ASCII Code Sequence

Decimal

(88)_D (P) (44)_D (Q1) (Q2)... (Qn) (44)_D (R1)
(R2)... (Rn) (13)_D

or (88)_D (P) (44)_D (Q1) (Q2)... (Qn) (44)_D (R1)
(R2)... (Rn) (00)_D

LPRINT "X 2, 254, 2
(1" long mark, tick at 1/2"

Hexadecimal

<58>_H <P> <2C>_H <Q1> <Q2>...
<Qn> <2C>_H <R1> <R2>... <Rn>
<0D>_H

or

<58>_H <P> <2C>_H <Q1> <Q2>...
<Qn> <2C>_H <R1> <R2> <R2>... <Rn>
<00>_H

Where (P) is the ASCII code for the parameter "p", (Q1) ... (Qn) for the characters of the value of "q" expressed in decimal form, and (R1) ... (Rn) for the characters of the digits of the value of "r" expressed in decimal.

N

(Mark)

Name

Mark

Format

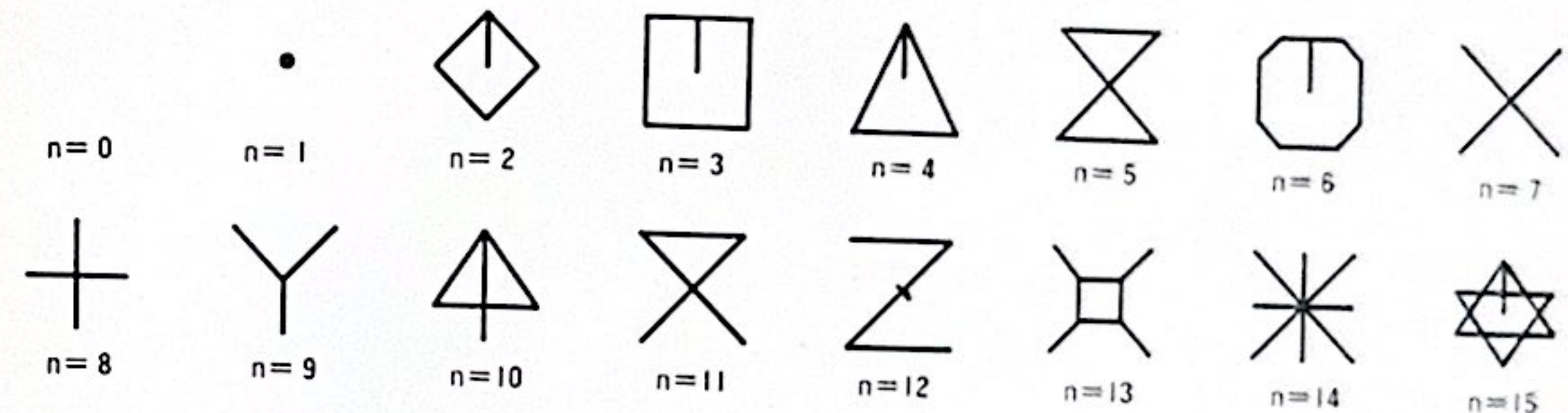
"N n" ▽

Function

To draw a mark at the current pen location.

Parameter type and range

The variable "n" specifies the mark according to the following table in the range 0 to 15.



The size and rotation of the mark can be specified using the "S" and "Q" commands respectively.

ASCII Code Sequence

Decimal

(78)_D (N1) (N2) (13)_D
or (78)_D (N1) (N2) (00)_D

Hexadecimal

<4E>_H <N1> <N2> <0D>_H
or <4E>_H <N1> <N2> <00>_H

Where (N1) and (N2) are the ASCII codes for the characters of "n".

Chapter 6 PRINTER MODE COMMANDS

6.1 Configuration Commands

ESC @

(Initialize)

Name Initialize Printer

Format CHR\$(27); "@";

Function To set the printer to the printer default state.

The HI-80 will not revert to the plotter mode, it will simply clear all printer parameters to their defaults in that mode.

Any data in the buffer will be lost.

ASCII code sequence

Decimal

(27)_D (64)_D

Hexadecimal

<1B>_H <40>_H